

APPENDIX C



Properties:

FieldToSearch

This property will return the name of the field currently being searched.

SearchType

This property will return a value indicating whether an Exact match or Subset of a word is chosen.

Methods:

DoSearch()

This method is a way to programmatically initiate the search.

DoCancel()

This method will close the Lookup window.

DoOK()

This method will close the Lookup window, firing the OnSearchSuccess event.

SetPrimaryField(Field Value)

This method will fill the primary criteria text field with the value.

SetSecondaryField(Field Value)

This method will fill the secondary criteria text field with the value.

SetSearchMode(Mode)

This method will determine the table that the lookup control will search. The values can be ICD or CPT.

Events:

SearchSuccess(Code, Short Description, Long Description)

This event is fired when the user clicks the OK button after successfully retrieving data.

CancelLookup()

This event is fired when the user clicks the Cancel button.

CRW Queue Module – Managing the quality assurance work flow queues

The Queue module is implemented as an OCX control. It contains two different windows. There is one window for choosing and presenting an existing queue query and one for creating a new query. The following are the methods and events for each window:

- Main Queue Window

This window will enable the user to retrieve a group of documents from which to choose a document to process.

Methods:

ShowQueue

This method shows the main queue window. This window has a combo box to choose an existing queue from. Below the combo box is a Grid from which to choose the document that the user wishes to process.

HideQueue

This method will hide the main queue window.

GetNextDoc

This method will retrieve the next document from the queue. The control will query the database to retrieve the most up to date result set.

GetPrevDoc

This method will retrieve the previous document from the queue. The control will query the database to retrieve the most up to date result set.

DoOpen()

Programmatically invoke the Open button functionality. This method opens the query and fires an event, `OpenQuery(Query String)` which passes the string contained in the query.

DoNew()

Invoke the New button functionality. This method opens the New Query window.

DoEdit()

Invokes the Edit button functionality. This method opens the New Query window the window and fires an event, `EditQuery()`.

GetColData(Row,Col)

This method enables the developer to programmatically extract the value from a specific row, column in the grid area of the control.

GetRowCount()

This method will return the number of rows that are currently in the result set.

Events:

OpenQuery(Query String)

This event is fired when the user clicks the Open button or if the `DoOpen` method is invoked.

OpenQueue (DocID)

This event is fired when the user double clicks on a document.

EditQuery()

This event is fired when the user clicks the Edit button or if the `DoEdit` method is invoked.

- New Query Window

This window will enable the user to create a new query script.

Methods:

DoCreate

Programmatically invoke the Create button functionality. This method creates the script and fires an event, `CreateNewQuery(Query String)` which passes the string contained in the query.

DoCancel

Invokes the Cancel button functionality. This method fires an event, `CancelNewQuery`.

AddField(FieldName, Value)

This method will add the field and its value to the query string.

Events:

CreateNewQuery(Query String)

This event is fired when the user clicks the Create button or if the `DoCreate` method is invoked.

CancelNewQuery()

This event is fired when the user clicks the Cancel button or if the `DoCancel` method is invoked.

The CCRWDoc object represents a hybrid of the CMasterDoc class, and other associated classes. The CCRWDoc object combines these together to provide a more 'natural' interface for use by applications that provide for user interaction. For example, the CCRWDoc COM object shelters the user (developer) from the details of network connections, rich text highlighting overhead, database connections and so on.

Methods:

Connect(ODBCDataSourceName)

Calls CMasterDoc, passing the ODBC datasource name from the CRW configuration file, to establish a connection to the master database. Returns success or failure status.

Disconnect()

Calls CMasterDoc to disconnection from the master database.

DbLogon(DbUserId, DbUserPassword)

Calls CMasterDoc, passing the database user ID and database password from the CRW configuration file, to establish a database session for the workstation. Returns success or failure status.

DbLogoff()

Calls to cancel a database session for the workstation.

SysLogon(UserId, Password)

Calls CUser, passing the user ID and password provided by the user, to establish a session with the data center system for the user. Returns success or failure status.

SysLogoff()

Calls CUser to a cancel a session with the data center system for the user.

GetSchemaVersion(MyVersion)

RequestsCUser to return the database schema version string, compares this to the version passed in and returns a status indicating that the versions are compatible or not.

Open(DocumentId)

Calls the CInpExpDoc, passing it the CMasterDoc to fill in and the requested document ID. The CInpExpDoc will return either with the document filled in and a success status, or a status indicating the document is locked, in which case the Open returns this status immediately to the caller. If the import was successful, calls the SetMode method of the CMasterDoc, requesting the mode be set to "EDIT", indicating that an editable copy of the document be created if it does not already exist.

Save()

Calls the CInpExpDoc, passing it the current CMasterDoc to export. Returns success or failure status.

Close()

Deletes the current CMasterDoc instance (effectively clearing it out), then creates a new CMasterDoc instance.

GetAllFields()

Calls the CMasterDoc requesting that all fields be returned with current data. The return result from this call is an two dimensional array of FieldId / Value combinations for every data field in a CRW document, including repeating fields such as ICD and CPT modifier fields. Document highlights are not returned at this point. Returns this array or NULL to indicate a failure.

GetOrigFields(FieldIdArray)

Calls the CMasterDoc requesting the original values for all fields listed in FieldIdArray (field values before any changes were made during the current edit session). FieldIdArray is a list of field names. CMasterDoc returns a list of values, with the position of these values corresponding to the position of its associated field name in the FieldIdArray. Returns this list or NULL to indicate failure.

GetICD9Fields()

Calls the CMasterDoc the current list of values for all ICD fields. CMasterDoc returns a list of Code / Desc values in Descending order. Returns this list or NULL to indicate failure.

GetOrigICD9Fields()

Calls the CMasterDoc requesting the original values for all ICD (field values before any changes were made during the current edit session). CMasterDoc returns a list of Code / Desc values in Descending order. Returns this list or NULL to indicate failure.

GetCPTFields()

Calls the CMasterDoc the current list of values for all CPT fields. CMasterDoc returns a list of Code / Desc values in Descending order. Returns this list or NULL to indicate failure.

GetOrigCPTFields()

Calls the CMasterDoc requesting the original values for all CPT (field values before any changes were made during the current edit session). CMasterDoc returns a list of Code / Desc values in Descending order. Returns this list or NULL to indicate failure.

GetCPTModFields()

Calls the CMasterDoc the current list of values for all CPT Modifier fields. CMasterDoc returns a list of Modifier / Lookup Value values in Descending order. Returns this list or NULL to indicate failure.

GetOrigCPTFields()

Calls the CMasterDoc requesting the original values for all CPT Modifiers (field values before any changes were made during the current edit session). CMasterDoc returns a list of Modifier / Lookup Value values in Descending order. Returns this list or NULL to indicate failure.

SetField(FieldId, Value)

Calls the CMasterDoc to update an individual field's data. Returns success or failure status.

SetICD9Field(Position, FieldId, Value)

Calls the CMasterDoc to update an individual ICD9 field's data. Position indicates an index into the list of ICD9 fields. FieldId indicates which field to update (ICDCode or ICDDesc). Returns success or failure status.

SetCPTField(Position, FieldId, Value)

Calls the CMasterDoc to update an individual CPT field's data. Position indicates an index into the list of CPT fields. FieldId indicates which field to update (CPTCode, CPTDesc, CrossRef, Modifiers, or LookupValues). Returns success or failure status.

MoveICD9(Position, Direction)

Calls the CMasterDoc to move the location of an individual ICD9 field within the list of ICD9 fields. Position indicates an index into the list of ICD9 fields. Direction is an enum requesting the selected field be moved UP or DOWN from its relative position in the list. Returns success or failure status.

MoveCPT(Position, Direction)

Calls the CMasterDoc to move the location of an individual CPT field within the list of CPT fields. Position indicates an index into the list of CPT fields. Direction is an enum requesting the selected field be moved UP or DOWN from its relative position in the list. Returns success or failure status.

MoveCPTMod(Position, Direction)

Calls the CMasterDoc to move the location of an individual CPT Modifier field within the list of CPT Modifier fields. Position indicates an index into the list of CPT Modifier fields. Direction is an enum requesting the selected field be moved UP or DOWN from its relative position in the list. Returns success or failure status.

DeleteICD9(Position, Direction)

Calls the CMasterDoc to delete an individual ICD9 field from the list of ICD9 fields. Position indicates an index into the list of ICD9 fields. Returns success or failure status.

DeleteCPT(Position, Direction)

Calls the CMasterDoc to delete an individual CPT field from the list of CPT fields. Position indicates an index into the list of CPT fields. Returns success or failure status.

DeleteCPTMod(Position, Direction)

Calls the CMasterDoc to delete an individual CPT Modifier field from the list of CPT Modifier fields. Position indicates an index into the list of CPT fields. Returns success or failure status.

GetHighlights(FieldId)

Requests the CMasterDoc to return the set of highlight start and end combinations for the given field. The highlighted combinations are returned in a two dimensional array of start and end offsets from the start of the Transcription text. Returns this array or NULL to indicate failure.